TCDMATH-11-09, ADP-11-23/T745

Polynomial Filtered HMC – an algorithm for lattice QCD with dynamical quarks

Waseem Kamleh[a,b], Mike Peardon[b]

[a]*Special Research Centre for the Subatomic Structure of Matter and Department of Physics, University of Adelaide 5005, Australia.*
[b]*School of Mathematics, Trinity College, Dublin 2, Ireland.*

arXiv:1106.5625v1 [hep-lat] 28 Jun 2011

Abstract

Polynomial approximations to the inverse of the fermion matrix are used to filter the dynamics of the upper energy scales in HMC simulations. The use of a multiple time-scale integration scheme allows the filtered pseudofermions to be evolved using a coarse step size. We introduce a novel generalisation of the nested leapfrog which allows for far greater flexibility in the choice of time scales. We observe a reduction in the computational expense of the molecular dynamics integration of between 3–5 which improves as the quark mass decreases.

1. Introduction

Studying the lattice representation of quantum chromodynamics (QCD) with light quarks remains a numerically intensive challenge, requiring large-scale computing resources. The quark fields in the path integral can not be manipulated easily on the computer and the most practical means of proceeding is to integrate them analytically and deal with the resulting fermion matrix determinant stochastically. After this step, a path integral over the gauge fields alone remains and since this is an ordinary integration problem, it can be tackled by the Monte Carlo method. At present, the best means of proceeding is to use an importance sampling technique, with a Markov chain of gauge field configurations being generated and used for subsequent stochastic estimation. The most widely used technique is the Hybrid Monte Carlo (HMC) algorithm [1]. Here, a fictitious time co-ordinate is introduced along with momentum degrees of freedom conjugate to the gauge fields and a Hamiltonian describing evolution in this new simulation time.

The difficulty with manipulating quark fields haunts this technique, making each application of the Markov process computationally intensive. Over the past ten years, the intensive use of Hybrid Monte Carlo in large-scale production runs means a good deal of practical experience has been gained. This experience has exposed the difficulties with approaching the physical quark mass and continuum limits of lattice QCD. Recent empirical observations [2] suggest the different topological sectors that contribute to the QCD vacuum are connected only weakly via the Markov transitions of most commonly used versions of HMC. A theoretical study of the approach to the continuum limit suggests that HMC is a non-renormalisable algorithm [3], which implies the computational cost of generating gauge field configurations with smaller lattice spacings is unpredictable.

These observations and unresolved questions strongly suggest that developing new ideas and modifications to algorithms such as HMC remains useful. There has been a great deal of activity over the past ten years developing the toolkit for numerical simulations, including Hasenbusch's mass preconditioning [4], RHMC [5], the use of different integrator schemes [6, 7, 8] and domain decomposition via the Schwartz alternating procedure [9, 10]. Some of these developments have enabled the first studies close to the theory parameters needed to make contact to physical data reliably [11]. There is still significant activity aiming to understand and improve HMC still further [12, 6]. A review of recent developments can be found in Ref. [13].

When using HMC, there are two obstacles to pushing down the quark mass in the simulation. Both stem from the fact that the fermion determinant must be represented stochastically using pseudofermions with a non-local action that features the inverse of the fermion matrix. The first hindrance is that the condition number of the fermion matrix increases at lighter quark mass, causing a large increase in the number of conjugate gradient iterations required to solve the linear system. Solving this system is required at each molecular dynamics step, which means the numerical expense of the inversions is the dominant cost for generating dynamical gauge fields. The second hindrance, an amplification of the first, is that the molecular dynamics step-size must be decreased as the quark mass is reduced in order to maintain control over the rapid fluctuations driven by the pseudofermion field.

Polynomial approximations to the inverse have been used to define fermion algorithms for some time [14, 15]. Previous explorations in the Schwinger model [16] showed that the introduction of a polynomial filter gave a cheap means of controlling the short-time-scale fluctuations in the molecular dynamics integration of HMC by introducing a separation of time scales in the molecular dynamics by directly factoring the pseudofermion action into multiple parts. In this paper, we describe the polynomial filter-

ing algorithm and present an investigation of its behaviour when applied to Monte Carlo studies of QCD. The paper is organised as follows: in Sec. 2 the method is described and some details of our implementation are discussed. Simulation results are presented in Sec. 3 and conclusions are drawn in Sec. 4

2. Polynomial Filtered Hybrid Monte Carlo Algorithm

The most powerful method for carrying out importance sampling Monte Carlo to estimate integrals over a large number of degrees of freedom q, of the form

$$\langle \mathcal{O} \rangle = \frac{1}{Z} \int \mathcal{D}q \;\; \mathcal{O}(q) \; e^{-S(q)}, \tag{1}$$

is to develop a Markov process with $\frac{1}{Z} e^{-S(q)}$ as its fixed-point probability distribution. Here, Z normalises this probability measure. The sequence of configurations generated by repeated applications of the process can then be used as an appropriate ensemble for importance-sampling estimation of $\langle \mathcal{O} \rangle$.

Hybrid Monte Carlo (HMC) is just such a Markov chain Monte Carlo technique, comprising of two component parts. New configurations of an extended system, (p, q) are proposed to a Metropolis test, and are accepted stochastically with probability

$$\mathcal{P}_{\rm acc} = \min(1, e^{-\Delta \mathcal{H}}), \tag{2}$$

based on the change in the Hamiltonian,

$$\mathcal{H}(p, q) = \frac{1}{2} \sum p^2 + S(q). \tag{3}$$

The extended system doubles the degrees of freedom, adding a conjugate variable p to each co-ordinate q. Since the Metropolis test ensures the probability of a configuration of this new system occurring in the Markov chain is given by $\exp\{-\mathcal{H}\} = \exp\{-T(p)\} \times \exp\{-S(q)\}$, the separable product of probability densities of p and q, these two sets of variables are independent random numbers. Subsequently, the new variables p can be discarded, leaving an ensemble of configurations of q with the desired probability distribution.

To ensure a useful acceptance probability for the Metropolis test, $\mathcal{H}$ is interpreted as the Hamiltonian describing the dynamics in the phase space generated by (p, q), where p is treated as a canonical momentum conjugate to q. In practical simulations, it is the molecular dynamics that requires most of the computer time for the Markov chain to evolve.

This construction strongly suggests that improving the efficiency of the algorithm requires both accelerating and improving the accuracy of the molecular dynamics integration process. As with many complex systems, the classical dynamics of the degrees of freedom of lattice QCD incorporates interactions with a broad range of characteristic time scales. Any attempt to improve the performance of molecular dynamics will be aided by describing a method that allows the separate dynamical scales to be treated differently. As a starting point, consider the multiple time-scale scheme of Sexton and Weingarten [17]. In this construction, an integrator which captures the different dynamical scales of different parts of the action can be defined. Integrators to evolve the system in the new time co-ordinate are constructed from two basic time-evolution operators, generated by kinetic and potential energy terms. Their effect on the system co-ordinates, (p, q) is

$$V_T(h) : \{p, q\} \longrightarrow \{p, q + h \; p\}, \tag{4}$$

and

$$V_S(h) : \{p, q\} \longrightarrow \{p - h \; \partial S, q\}, \tag{5}$$

where ∂S is the "extended force" due to the action S. The simplest time-reversible integrator is then built from the leap-frog scheme,

$$V(h) = V_S(\frac{h}{2}) V_T(h) V_S(\frac{h}{2}), \tag{6}$$

and repeated applications of this building block evolves the system with the Hamiltonian conserved up to corrections of $\mathcal{O}(h^2)$.

2.1. Multiple time-scales in molecular dynamics integrators

If the action and thus the Hamiltonian is split into two parts H_1 and H_2,

$$\mathcal{H} = \underbrace{\frac{1}{2} \sum p^2 + S_1(q)}_{H_1} + \underbrace{S_2(q)}_{H_2}, \tag{7}$$

where S_1 is an action capturing the high-frequency modes and S_2 captures the low-frequency ones, then a generalised leap-frog integrator can be written. The two leap-frog integrators for the two Hamiltonians are defined as

$$V_1(h) = V_{S_1}(\frac{h}{2}) V_T(h) V_{S_1}(\frac{h}{2}), \tag{8}$$

$$V_2(h) = V_{S_2}(h), \tag{9}$$

and a compound integrator for the full Hamiltonian can be constructed by combining the two components:

$$V(h) = V_2(\frac{h}{2}) \left[V_1(\frac{h}{m}) \right]^m V_2(\frac{h}{2}), \tag{10}$$

where $m \in \mathbb{N}$. This compound integrator effectively introduces two time-scales into the evolution, h and h/m.

Since the force term for S_1 must be evaluated many more times than that for S_2, the method will only be useful if two conditions are met simultaneously. First, as already suggested, the two actions must effectively split the dynamical scales into high and low-frequency parts in S_1 and S_2, and also, the evaluation of ∂S_1 must be computationally much cheaper than that for ∂S_2.

2.2. HMC for Lattice QCD

When using HMC to generate an ensemble of gauge configurations for importance sampling Monte Carlo estimation of the path integral of lattice QCD, a few details arise. The first is that the integration variables are elements of the gauge group, $SU(3)$, and the gauge invariant Haar measure must be used. This can be achieved by making the conjugate momenta elements of the Lie algebra of the gauge group, and by modifying the action of V_T slightly;

$$V_T(h) : (P, U) \longrightarrow (P, e^{ihP}U). \tag{11}$$

Correspondingly, the effect of the evolution operator V_S must then be

$$V_S(h) : (P, U) \longrightarrow (P - h\Sigma, U), \tag{12}$$

where

$$\Sigma = \frac{\Lambda - \Lambda^\dagger}{2i} - \frac{1}{N}\mathrm{Im}\mathrm{Tr}\Lambda, \tag{13}$$

with

$$\frac{dS}{dt} = \mathrm{Re}\mathrm{Tr}\left\{\Lambda\frac{dU}{dt}\right\}. \tag{14}$$

While this defines molecular dynamics on the group manifold, the action for QCD with the quark field dynamics included requires some manipulation. After integrating out the Grassmann variables representing the quark fields, the importance sampling probability measure for two degenerate flavours of quarks becomes

$$\mathcal{P}[U] = \frac{1}{Z}\ \det{}^2 M[U]\ \ e^{-S_G[U]} = \frac{1}{Z}e^{-S_{\mathrm{eff}}[U]}. \tag{15}$$

The effective action for this measure is given by

$$S_{\mathrm{eff}}[U] = -2\mathrm{Tr}\ \log M[U]\ \ + S_G[U]. \tag{16}$$

Unfortunately, evaluating this action and the force terms that arise from the molecular dynamics is very cumbersome on the computer. A solution requires the introductions of "pseudofermions", which represent the fermion determinant as a Gaussian integral;

$$\det{}^2 M[U] = \int \mathcal{D}\phi\mathcal{D}\phi^*\ \ e^{-\phi^*(M^\dagger M)^{-1}\phi}, \tag{17}$$

and the new effective action for the system is

$$S_{\mathrm{eff}}[U, \phi, \phi^*] = \phi^* K^{-1}\phi + S_G[U], \tag{18}$$

where $K = M^\dagger M$. The γ_5-Hermiticity property of the fermion matrix has been exploited here; since $M^\dagger = \gamma_5 M \gamma_5$, $\det M^\dagger = \det M$ and hence $\det K = \det{}^2 M$. The molecular dynamics force term for the pseudofermions can now be computed; this requires the application of the inverse of the fermion matrix. While this step is computationally very expensive, it is at least tractable. Since the action has two components, it seems natural to use the Sexton-Weingarten scheme to integrate both parts with different time-scales appropriate to their dynamics. Unfortunately, at light quark masses the high-frequency dynamics occurs in the pseudofermion action, which also has the most expensive force evaluation.

2.3. The polynomial filtered hybrid Monte Carlo algorithm

The requirement that the integrator for the shortest time-step has a force term that is computationally cheap to implement leads to using low-order polynomial approximations to the inverse of the fermion matrix. Clearly, a short polynomial satisfies the condition that its force is cheap to evaluate. The hope is that such a simple approximation might be able to mimic the short-distance part of the propagator, and so capture the high-frequency dynamics expected in this part of the system.

Making use of the identity which holds for any polynomial, $\mathcal{P}$ of the (two flavour) fermion matrix

$$\det K = \frac{\det K\mathcal{P}(K)}{\det \mathcal{P}(K)}, \tag{19}$$

suggests introducing two auxiliary integrals for each of these determinants separately,

$$\det{}^2 M = \int \mathcal{D}\phi\mathcal{D}\phi^*\mathcal{D}\chi\mathcal{D}\chi^*\ \ e^{-\phi^*(\mathcal{P}K)^{-1}\phi - \chi^*\mathcal{P}\chi}. \tag{20}$$

The action for QCD then becomes

$$S_{\mathrm{filter}}[U] = \phi^*(\mathcal{P}K)^{-1}\phi + \chi^*\mathcal{P}\chi + S_G[U]. \tag{21}$$

This action is separated into two parts, S_1 and S_2,

$$S_1 = \phi^*(\mathcal{P}K)^{-1}\phi, \tag{22}$$

$$S_2 = \chi^*\mathcal{P}\chi + S_G[U]. \tag{23}$$

In the limit that a perfect representation of the inverse is constructed, *i.e.* $K\mathcal{P}(K) \to I$, the first action becomes independent of the gauge fields and so induces no molecular dynamics force. Conversely, as the order of the polynomial reduces, $\mathcal{P} \to I$ and then the system reproduces the pseudofermion action of Eqn. (18). The polynomial can thus be used to capture as much or as little of the dynamics of the fermion action as is necessary to extract the high-frequency modes as cheaply as possible.

2.4. Adding additional time-scales.

The determinant ratio identity of Eqn. (19) can be extended to make use of more than one polynomial

$$\det K = \det K\mathcal{P}_2(K) \times \frac{1}{\det \mathcal{P}_1^{-1}(K)\mathcal{P}_2(K)} \times \frac{1}{\det \mathcal{P}_1(K)} \tag{24}$$

The order of the first polynomial, n_1 is chosen to be much less than that of the second, $n_2 \gg n_1$. For this construction to be useful, the two polynomials should be constructed so all the roots of the first are contained in the set of roots of the second. In this case, $\mathcal{P}_2$ can be written as the product of two polynomials,

$$\mathcal{P}_2 = \mathcal{P}_1\mathcal{Q} \tag{25}$$

where the order of the new polynomial $\mathcal{Q}$ is $n_q = n_2 - n_1$. Now, the fermion determinant can be re-expressed using

three integral representations, and the resulting action for QCD becomes

$$S_{2-\mathrm{poly}} = \phi^*(\mathcal{P}_2 K)^{-1}\phi + \chi_2^*\mathcal{Q}^\dagger\mathcal{Q}\chi_2 + \chi_1^*\mathcal{P}_1^\dagger\mathcal{P}_1\chi_1 + S_G. \tag{26}$$

Judicious choice of the two polynomials, $\mathcal{P}_1$ and $\mathcal{P}_2$ should capture the dynamics of successively increasing time-scales, and allow a generalised version of the Sexton-Weingarten scheme to be employed.

2.5. Polynomial approximation to $\frac{1}{z}$

Naturally, in order to be able to perform Monte Carlo simulations $\mathcal{P}(K)$ must be a real polynomial and hence possess roots in complex conjugate pairs. In this work we use a Chebyshev approximation to to $1/z$,

$$P_n(z) = a_n \prod_{k=1}^{n}(z - z_k) \approx \frac{1}{z}. \tag{27}$$

The roots are given by [14, 18]

$$z_k = \mu(1 - \cos\theta_k) - i\sqrt{\mu^2 - \nu^2}\sin\theta_k, \tag{28}$$

with $\theta_k = \frac{2\pi k}{n+1}$. The normalisation is then

$$a_n = \frac{1}{\mu\prod_{k=1}^{n}(\mu - z_k)}. \tag{29}$$

The polynomial possesses roots which lie on an ellipse in the complex plane. The parameters μ and ν should be chosen such that the spectrum of K lies within the ellipse. Within this constraint one is free to tune the two parameters. For the choice $\mu = 1, \nu = 0$ the polynomial coincides with that given by the truncation of the Taylor series expansion of $1/z$ about $z = 1$.

2.6. A generalised multi-scale leap-frog integrator

For a two flavour simulation with up to two polynomial terms in the Hamiltonian, we introduce the possibility of simulating at four different time scales, one for each of the terms involved,

$$S_1 = S_G, \qquad S_2 = \chi_1^*\mathcal{P}_1\chi_1,$$
$$S_3 = \chi_2^*\mathcal{Q}\chi_2, \qquad S_4 = \phi_{2\mathrm{f}}^*(\mathcal{P}_2 K)^{-1}\phi_{2\mathrm{f}}.$$

Associate a timestep h_i to each term S_i and a corresponding integer N_i such that $h_i = 1/N_i$. h_1 corresponds to the step-size at which the gauge fields are updated. Assuming that $N_i > N_j$ for $i < j$, the Sexton-Weingarten nested leapfrog algorithm then requires that $N_i | N_{i-1}\ \forall\, i > 1$. The restrictions this places on the various N_i may not be the most efficient or flexible way of performing the molecular dynamics integration.

A generalised scheme [19] in which the only requirement is

$$N_i | N_1\ \forall\ i > 1, \tag{30}$$

can be defined. In a standard leapfrog algorithm, one alternates between updates V_T to the gauge field U and updates V_S to the conjugate momenta. Let V_i denote the update to P corresponding to the action S_i. Now, as the guide bosons are held fixed during an integration the updates V_i only depend upon the gauge field. As the updates V_i are additive to P, it follows that the different V_i commute:

$$V_i(\frac{h_i}{2})V_j(h_j)V_i(\frac{h_i}{2}) = V_j(h_j)V_i(h_i) = V_i(h_i)V_j(h_j). \tag{31}$$

Define the integers

$$m_i = N_1 \div N_i \tag{32}$$

to be the ratios of the scales. In order to construct our reversible integrator we first define a map

$$\Theta[V; m, k \in \mathbb{N}] = \begin{cases} V \text{ if } m|k \\ I(\text{the identity}) \text{ otherwise.} \end{cases} \tag{33}$$

Let m_T be the lowest common multiple of $\{m_i\}$, and let h_T be the smallest time step (in our case $h_T = h_1$). Then our integrator is

$$V(h) = \prod_i V_i(\frac{h_i}{2})\times \prod_{k=1}^{m_T-1} V_T(h_T)\Big\{\prod_i \Theta[V_i(h_i), m_i, k]\Big\} \times V_T(h_T)\prod_i V_i(\frac{h_i}{2}), \tag{34}$$

where $h = m_T h_T$ is the total timestep taken by V. While it appears cumbersome here, the above expression is straightforwardly implemented in software. We demonstrate this with a pseudocode implementation here. Denote by $\{a \equiv b \mod m\}$ the usual notion of congruence modulo m. Then we can implement the generalised integrator as follows:

- For each term in the action S_i perform an initial half-step $V_i(\frac{1}{2}h_i)$ updating P.
- Loop over $j = 1$ to $N - 1$
  - Apply $V_T(h)$ to update U.
  - If $\{0 \equiv j \mod m_i\}$ apply $V_i(h_i)$ to update P
- Apply $V_T(h)$ to update U.
- For each term in the action S_i perform a final half-step $V_i(\frac{1}{2}h_i)$ updating P.

The advantage of the generalised integrator is that it allows finer control when there are many different scales. An analysis of the finite-step size errors for the generalised integrator is provided in Appendix A.

2.7. Polynomial force term

For completeness, we review here the molecular dynamics force generated by the polynomial terms [15, 20]. Given a polynomial

$$\mathcal{P}(K) = a_n \prod_{i=1}^{n} (K - z_i) \tag{35}$$

and the following action

$$S_P = \phi^\dagger \mathcal{P}(K) \phi, \tag{36}$$

we define the auxiliary fields

$$\chi_j = \prod_{i>j} (K - z_i)\phi, \tag{37}$$

$$\eta_j = a_n \prod_{i<j} (K - z_i^*)\phi, \tag{38}$$

Then we have

$$\frac{dS_P}{dt} = \mathrm{Tr} \left\{ \sum_{j=1}^{n} \left(\eta_j^\dagger \frac{dK}{dt} \chi_j \right) \right\}. \tag{39}$$

Although the above equation appears to require $\mathcal{O}(n^2)$ matrix operations one can reduce this to $\mathcal{O}(n)$ by trading off for storage of $\mathcal{O}(n)$ intermediate fields [15].

3. Simulation Results

Results are calculated on lattices using the Wilson gauge action and the unimproved even-odd preconditioned Wilson fermion action. In order provide a straightforward comparison with alternative algorithms, we chose physical parameters that have been used elsewhere [10, 21]. The gauge coupling is set to $\beta = 5.6$. We work at two quark masses, $\kappa = 0.1575$ and 0.15825 which correspond to pion masses of 600 and 400 MeV respectively.

We begin by examining the effectiveness of a single polynomial filter ($n_q = 0$). We measure the size of the force Σ_F due to the filtered pseudofermion term S_F and of the force Σ_P due to the polynomial term S_P. Each element of the force is in the Lie algebra $su(3)$ hence as in other work[22] we use the Lie norm $||X||^2 = -2\,\mathrm{Tr}\, X^2$. Across a configuration we measure the mean and maximum values of $||\Sigma||$. These values do not vary much from one trajectory to the next. Tuning of the parameters μ and $\nu < \mu$ is done by choosing the values which minimise a sample measurement of the force on a single configuration.

The left-hand plot in Figure 1 shows typical mean and maximum forces due to the (filtered) pseudofermions. We see that even using very short polynomials gives significant reduction in the force, which continues to decrease as one increases the order of the polynomial n_p. We note however that the rate of the reduction in the force is sub-linear in n_p.

κ	a	b	c	χ^2/dof
0.1575	0.0150(4)	0.0778(5)	0.581(2)	2.12
0.15825	0.0271(9)	0.0187(9)	0.586(1)	15.8

Table 1: Values obtained for the fit parameters for $\tau_0 = a + bn_1^c$.

The right-hand plot in Figure 1 shows typical mean and maximum forces due to the polynomial term. We see that this is roughly independent of the size of the polynomial, and is approximately equal to the value for $n_p = 0$. This indicates the the size of the force is dominated by the shortest scale present, which strongly supports the motivation for separating the time scales.

We can also measure the efficacy of the polynomial filter by considering the acceptance rate as a function of the pseudofermion step size. Figure 2 shows the acceptance rate as a function of h_F for a single polynomial filter (of various order n_p). The polynomial step size was kept approximately fixed. Fits are performed using the complementary error function

$$\rho_{\mathrm{acc}} = \mathrm{erfc}(\frac{h_F^2}{\tau_0^2}). \tag{40}$$

We call τ_0 the characteristic scale. The characteristic scale as a function of polynomial order is shown in the right hand plot of Figure 2. To gain some intuitive understanding, we can see that using a step size of half the characteristic scale will yield an acceptance rate of approximately 0.7. We fit the characteristic scale as function of n_1 as follows

$$\tau_0 = a + bn_1^c. \tag{41}$$

The addition of a second polynomial filter does not change the qualitative behaviour of the quantities we have examined. To study the effects of adding a second polynomial, we fix $n_p = 4$ and vary n_q. Figure 3 shows the size of the force Σ_F due to the pseudofermions and that due to the second polynomial term Σ_Q. Even at moderate n_q, $||\Sigma_F||$ is significantly reduced compared to its unfiltered value, by a factor of between 10-20. We see again that $||\Sigma_F||$ decreases sub-linearly with n_q and that $||\Sigma_Q||$ is essentially independent of n_q, being dominated by the shortest scale present. Figure 4 shows the acceptance rate as a function of h_F for a 2-filter system (with $n_p = 4$ and various n_q) on the left, and the characteristic scale τ_0 as a function of $n_2 = n_p + n_q$ on the right. We again perform a fit for τ_0 as before.

We repeat the comparison of force sizes, acceptance rates and characteristic scales for our lighter quark mass at $\kappa = 0.15825$. Results are shown in Figures 5 and 6 for a variety of polynomial filters. We see the same qualitative behaviour in these results as those for $\kappa = 0.1575$ discussed above. In the right hand plot of Figure 6 fits for the characteristic scale for both masses using combined data for the single and dual polynomial filter results are shown. The values obtained for a, b, c are given in Table 1.

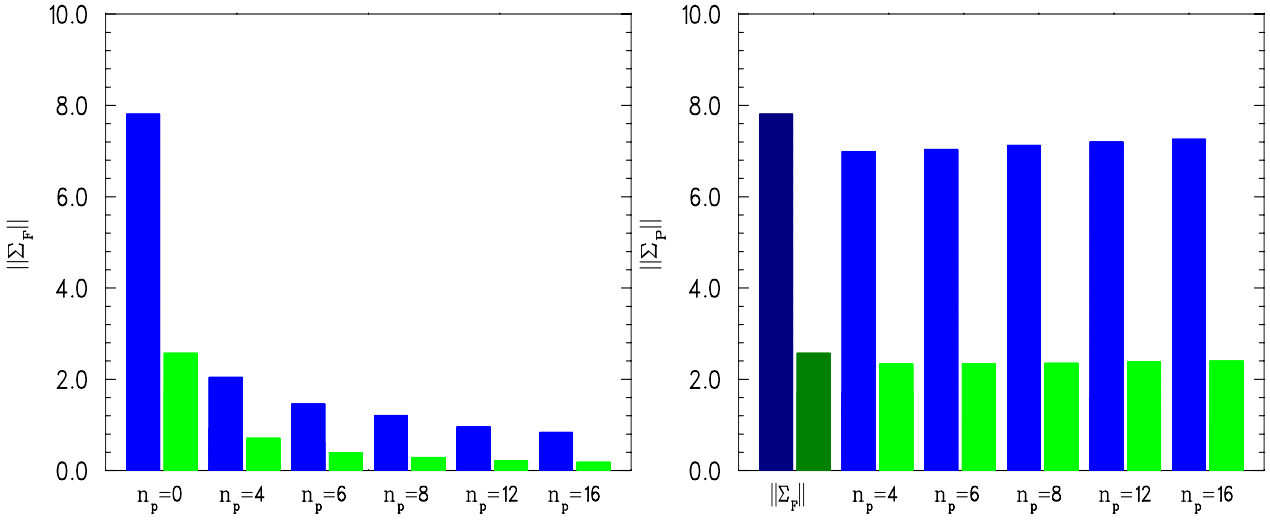

Figure 1: The size of the force due to the pseudofermions $||\Sigma_F||$ (left) and that due to the polynomial term $||\Sigma_P||$ (right), as a function of (single) polynomial filter order n_p. The maximum and average force size is shown for $\kappa = 0.1575$. For comparison, the values for $||\Sigma_F||$ with $n_p = 0$ are also shown as the leftmost darker coloured bars in the graph for $||\Sigma_P||$ (right).

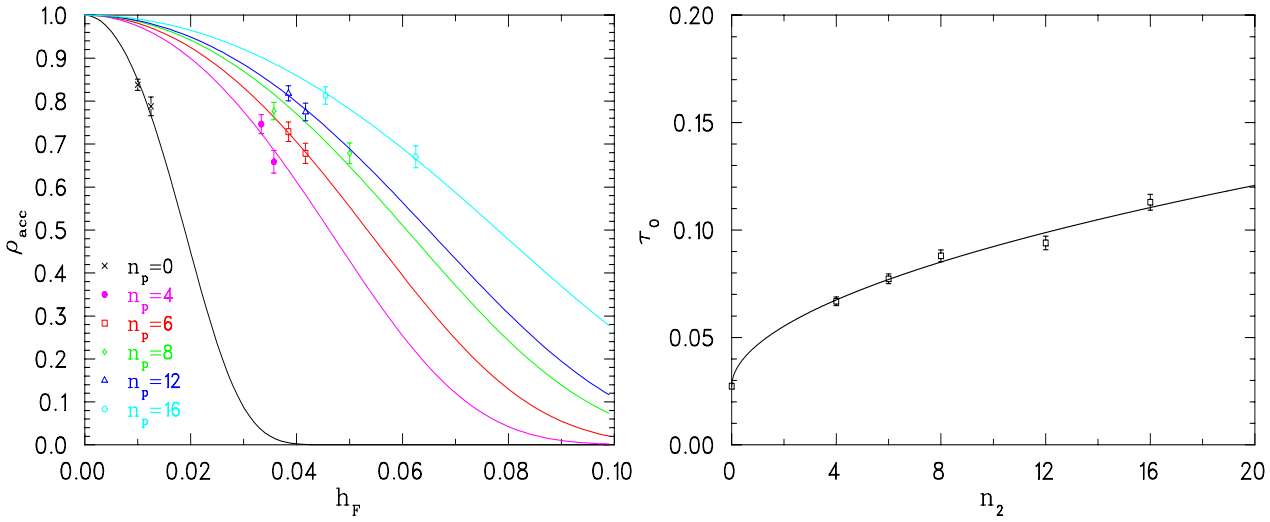

Figure 2: (Left) Trajectory acceptance rate as a function of pseudo-fermion step size for a single polynomial filter of various order n_p. (Right) Fit of the characteristic scale τ_0 as a function of polynomial filter order $n_2 = n_p$. Results are shown for $\kappa = 0.1575$.

In order to measure the relative numerical cost of the different choices for n_p, n_q we calculate the total number of Dirac matrix applications needed per molecular dynamics trajectory at an acceptance rate of 0.7, and we denote this value by $D_{0.7}$. Inverting Eq. 40 allows us to determine the pseudofermion step size h_F that corresponds to $\rho_{\rm acc} = 0.7$. Keeping the polynomial step sizes approximately fixed at $N_P \approx 120, N_Q \approx 30$, it is then a simple matter to calculate $D_{0.7}$. Figure 7 shows $D_{0.7}$ as a function of polynomial order n_2. We see that although there is a large initial reduction in cost, $D_{0.7}$ becomes nearly flat after about $n_2 = 4$. The reason for this is that our polynomial of order n_2 reduces the condition number of the Dirac matrix by less than a factor of $(n_2+1)^{-1}$. As each iteration of the conjugate gradient routine requires n_2+1 applications of the Dirac matrix, we can see that the cost of inverting the filtered Dirac matrix is increasing. This is offsetting the gain we obtain by having to invert less often. A potential improvement which could prove particularly useful for larger polynomials is to make use of a linear multi-shift inverter[23, 24, 25] to evaluate the action of the inverse of $M\mathcal{P}$.

However, at large n_2 the force contribution from the fermions is very small indeed. This allows us to slacken the inversion target residual during the molecular dynamics integration. Slackening the residual from 10^{-7} to 10^{-4} significantly reduces $D_{0.7}$. This is equivalent to working with an approximate Hamiltonian during the integration, and hence may put an upper bound less than one on the acceptance rate. This effect is observed for example if we try $r = 10^{-3}$.

As an aside, we also performed some tests using a variant polynomial filter applied to the Wilson operator M

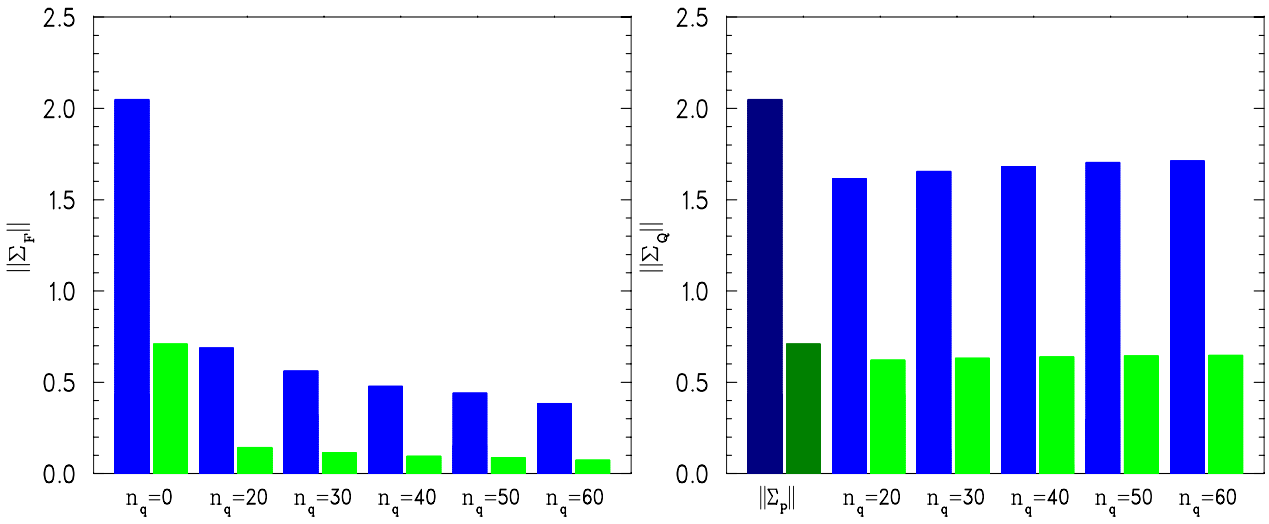

Figure 3: The size of the force due to the pseudofermions $||\Sigma_F||$ (left) and that due to the polynomial term $||\Sigma_Q||$ (right), as a function of polynomial filter order n_q with $n_p = 4$ fixed. The maximum and average force size is shown for $\kappa = 0.1575$. For comparison, the values for $||\Sigma_P||$ with $n_p = 4, n_q = 0$ are also shown as the leftmost darker coloured bars in the graph for $||\Sigma_Q||$ (right).

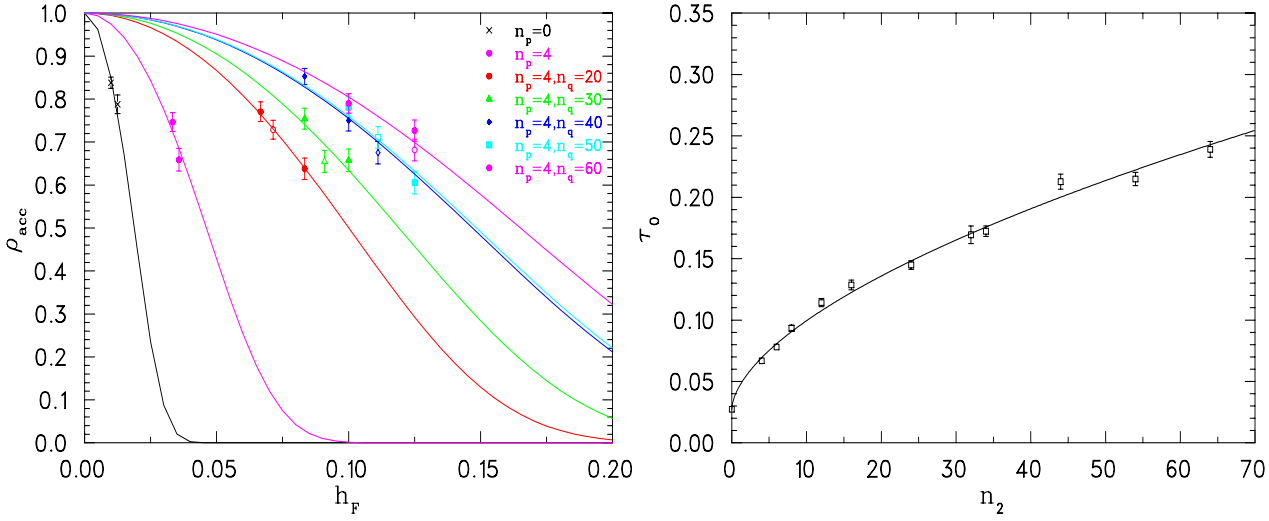

Figure 4: (Left) Trajectory acceptance rate as a function of pseudo-fermion step size for a dual polynomial filter of fixed order $n_p = 4$ and various order n_q. (Right) Fit of the characteristic scale τ_0 as a function of polynomial filter order $n_2 = n_p + n_q$. Results are shown for $\kappa = 0.1575$.

directly rather than $K = M^\dagger M$. However, as one goes to light quark masses we find that the spectrum of M (which is complex) intrudes upon the boundary of the elliptical region defined by the roots of the polynomial, making this alternative method unfeasible.

4. Conclusions

The use of polynomials to separate different time scales in the molecular dynamics integration step of the HMC algorithm applied to lattice QCD with dynamical quark fields was introduced. This shifts the most expensive force terms to the coarsest scale (and vice versa), allowing multiple time scale integration schemes to be effective. The procedure is extensible allowing not only the separation of UV and IR dynamics, but also the introduction of intermediate scales. A novel generalisation of the leapfrog integrator was introduced which allows for far greater flexibility in choosing the time scale that one associates with the dynamics induced by a particular term in the action. The integrator is applicable to any simulation that makes use of multiple time scales.

Polynomial approximations to the inverse were shown to be successful UV filters for two flavour simulations. One possible improvement is to use a multi-shift solver to calculate the inversion of $M\mathcal{P}$. This should prove most advantageous when dealing with polynomials of larger order, which occur when using an intermediate filter. The use of a polynomial filter can be applied to single flavour simulations using a variety of implementations. A detailed descrip-

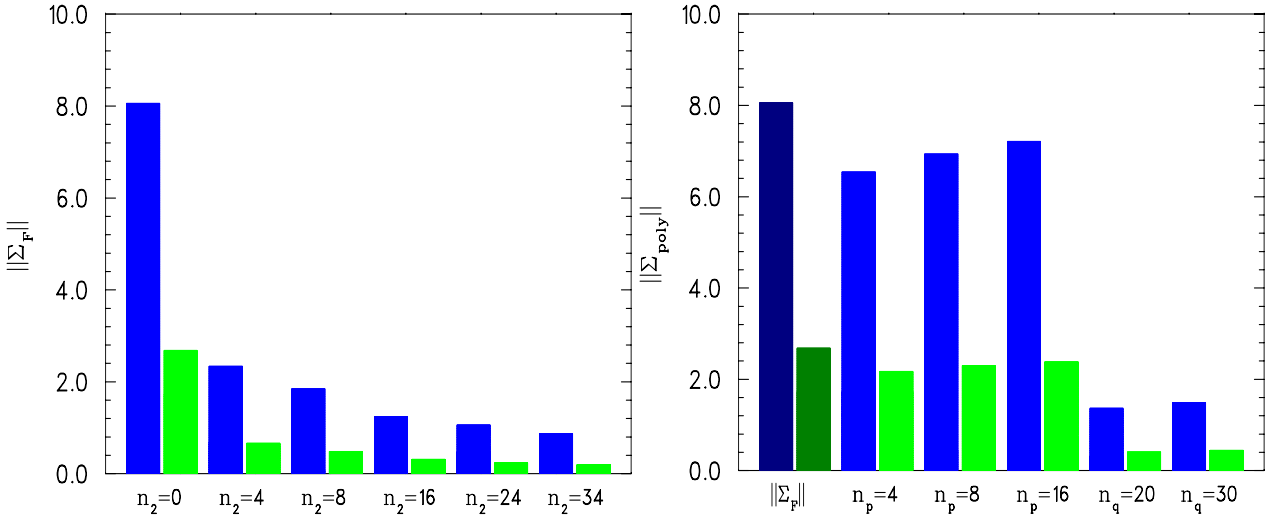

Figure 5: The size of the force due to the pseudofermions $||\Sigma_F||$ (left) and that due to the polynomial term $||\Sigma_{poly}||$ (right), as a function of polynomial filter order. Results are shown for various n_p, with two results for $n_q = 20, 30$ for which $n_p = 4$. The maximum and average force size is shown for $\kappa = 0.15825$. For comparison, the values for $||\Sigma_F||$ with $n_p = 0, n_q = 0$ are also shown as the leftmost darker coloured bars in the graph for $||\Sigma_{\rm poly}||$ (right).

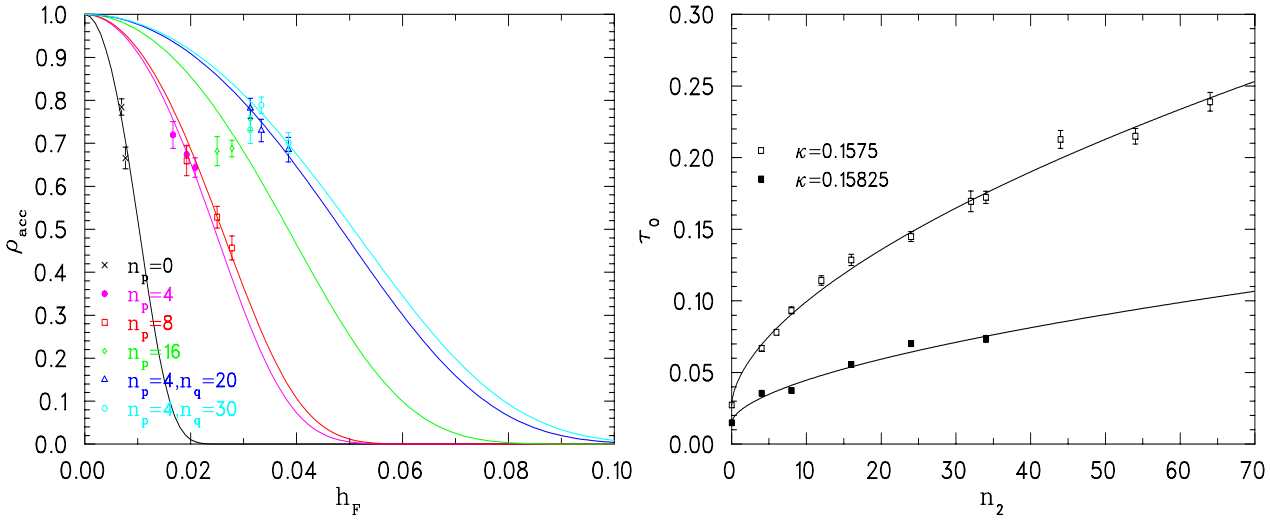

Figure 6: (Left) Trajectory acceptance rate as a function of pseudo-fermion step size for a dual polynomial filter of various order n_p and n_q, with $\kappa = 0.1575$. (Right) Fit of the characteristic scale τ_0 as a function of polynomial filter order $n_2 = n_p + n_q$. Combined results are shown for both $\kappa = 0.1575$ and $\kappa = 0.15825$.

tion and comparison of these techniques are the subject of future work.

5. Ackowledgements

We thank the Trinity Centre for High-Performance Computing (TCHPC) and the Australian Partnership for Advanced Computing (APAC) for providing access to the computational resources used for this work. The research was supported by Science Foundation Ireland under grants 04/BR/P0266 and 07/RFP/PHYF168.

Appendix A. Integrator Analysis

We perform an error analysis of our generalised integrator for a simple choice of stepsizes, following the procedure in [17]. Given a Hamiltonian $\mathcal{H}$ we can write the evolution operator for our system as $\exp h\hat{\mathcal{H}}$, with stepsize h. Here we have defined $\hat{\mathcal{H}}$ as the linear operator on the vector space of functions f on phase space (p, q) defined by the Poisson bracket

$$\hat{\mathcal{H}} f = -\{\mathcal{H}, f\} = \sum_i \left(\frac{\partial \mathcal{H}}{\partial p_i} \frac{\partial f}{\partial q_i} - \frac{\partial \mathcal{H}}{\partial q_i} \frac{\partial f}{\partial p_i} \right). \qquad \text{(A.1)}$$

If we write the Hamiltonian as

$$\mathcal{H} = T + S_1 + S_2 + S_3 + S_4 + \ldots \qquad \text{(A.2)}$$

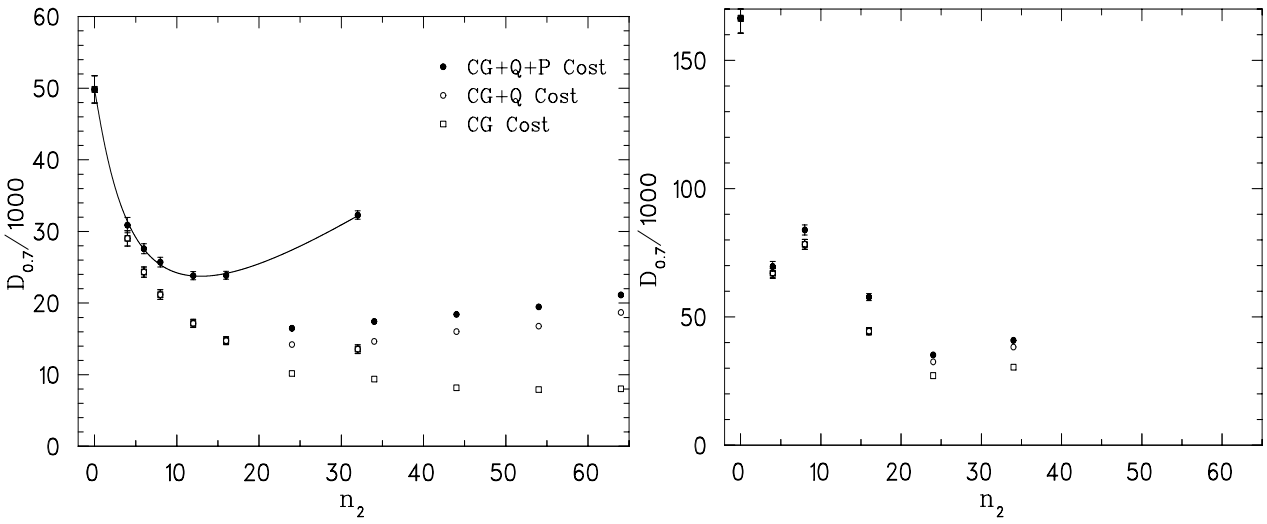

Figure 7: The cost $D_{0.7}$ as a function of polynomial filter size n_2, for $\kappa = 0.1575$ (Left) and $\kappa = 0.15825$ (Right).

then for each term in the Hamiltonian we can correspondingly define a linear operator using the Poisson bracket relation above.

Proceeding with the error analysis, we make use of the Baker-Campbell-Hausdorff result,

$$e^{\lambda\hat{A}}e^{\lambda\hat{B}}e^{\lambda\hat{A}} = \exp\Big(\lambda(2\hat{A}+\hat{B})+ \frac{\lambda^3}{6}([[\hat{A},\hat{B}],\hat{A}]+[[\hat{A},\hat{B}],\hat{B}]) + O(\lambda^4)\Big) \quad \text{(A.3)}$$

and apply this to the generalised leapfrog integrator in the simple case of $\mathcal{H} = T + S_1 + S_2$, where the time scale for each term in $\mathcal{H}$ corresponds to a number of integration steps $N_T = 6, N_1 = 3$ and $N_2 = 2$ respectively. The integrator for this simplest non-trivial case can be written as

$$e^{\frac{h}{4}\hat{S}_2}e^{\frac{h}{6}\hat{S}_1}e^{\frac{h}{3}\hat{T}}e^{\frac{h}{3}\hat{S}_1}e^{\frac{h}{6}\hat{T}}e^{\frac{h}{2}\hat{S}_2}e^{\frac{h}{6}\hat{T}}e^{\frac{h}{3}\hat{S}_1}e^{\frac{h}{3}\hat{T}}e^{\frac{h}{6}\hat{S}_1}e^{\frac{h}{4}\hat{S}_2}. \quad \text{(A.4)}$$

Repeated application of our BCH result allows us to deduce that the above expression can be written as

$$\exp\Big(h\hat{\mathcal{H}} + h^3\big(\frac{1}{48}[[\hat{S}_2,\hat{T}],\hat{T}] + \frac{1}{96}[[\hat{S}_2,\hat{T}],\hat{S}_2] + \frac{1}{216}[[\hat{S}_1,\hat{T}],\hat{S}_1] + \frac{1}{108}[[\hat{S}_1,\hat{T}],\hat{T}] + \frac{1}{36}[[\hat{S}_1,\hat{T}],\hat{S}_2]\big)\Big) \quad \text{(A.5)}$$

From this expression we can immediately see that the error in the generalised integrator relative to the leading term is $O(h^2)$, just as for the regular leapfrog.

If we examine the individual leapfrog integrators corresponding to

$$H_1 = T + S_1, \quad H_2 = T + S_2 \quad \text{(A.6)}$$

we obtain

$$e^{\frac{h}{6}\hat{S}_1}e^{\frac{h}{3}\hat{T}}e^{\frac{h}{3}\hat{S}_1}e^{\frac{h}{3}\hat{T}}e^{\frac{h}{3}\hat{S}_1}e^{\frac{h}{3}\hat{T}}e^{\frac{h}{6}\hat{S}_1} = \exp\Big(h\hat{H}_1 + h^3\big(\frac{1}{108}[[\hat{S}_1,\hat{T}],\hat{T}] + \frac{1}{216}[[\hat{S}_1,\hat{T}],\hat{S}_1]\big)\Big), \quad \text{(A.7)}$$

and

$$e^{\frac{h}{4}\hat{S}_2}e^{\frac{h}{2}\hat{T}}e^{\frac{h}{2}\hat{S}_2}e^{\frac{h}{2}\hat{T}}e^{\frac{h}{4}\hat{S}_2} = \exp\Big(h\hat{H}_2 + h^3\big(\frac{1}{48}[[\hat{S}_2,\hat{T}],\hat{T}] + \frac{1}{96}[[\hat{S}_2,\hat{T}],\hat{S}_2]\big)\Big) \quad \text{(A.8)}$$

Hence we see that the only difference between the individual integrators and our generalised integrator is the cross term $[[\hat{S}_1,\hat{T}],\hat{S}_2]$. The algorithm is identical to a standard nested leapfrog in the case where $N_i | N_{i-1}$.

References

[1] Duane, S., Kennedy, A. D., Pendleton, B. J., and Roweth, D., Phys. Lett. **B195** (1987) 216.
[2] Schaefer, S., Sommer, R., and Virotta, F., PoS **LAT2009** (2009) 032.
[3] Lüscher, M. and Schaefer, S., JHEP **04** (2011) 104.
[4] Hasenbusch, M., Phys. Lett. **B519** (2001) 177.
[5] Clark, M. A. and Kennedy, A. D., Phys. Rev. Lett. **98** (2007) 051601.
[6] Kennedy, A. D., Clark, M. A., and Silva, P. J., (2009).
[7] Chin, S. A. and Kidwell, D. W., Phys. Rev. **E62** (2000) 8746.
[8] Omelyan, I. A., Mryglod, I. M., and Folk, R., Phys. Rev. **E66** (2002) 026701.
[9] Lüscher, M., JHEP **05** (2003) 052.
[10] Lüscher, M., Comput. Phys. Commun. **165** (2005) 199.
[11] Aoki, S. et al., Phys. Rev. **D81** (2010) 074503.
[12] Lüscher, M., Commun. Math. Phys. **293** (2010) 899.
[13] Jung, C., PoS **LAT2009** (2009) 002.
[14] Lüscher, M., Nucl. Phys. **B418** (1994) 637.
[15] Frezzotti, R. and Jansen, K., Phys. Lett. **B402** (1997) 328.
[16] Peardon, M. J. and Sexton, J., Nucl. Phys. Proc. Suppl. **119** (2003) 985.

[17] Sexton, J. C. and Weingarten, D. H., Nucl. Phys. **B380** (1992) 665.
[18] Borici, A. and de Forcrand, P., Nucl. Phys. **B454** (1995) 645.
[19] Kamleh, W., AIP Conf. Proc. **1354** (2011) 58.
[20] Frezzotti, R. and Jansen, K., Nucl. Phys. **B555** (1999) 395.
[21] Urbach, C., Jansen, K., Shindler, A., and Wenger, U., Comput. Phys. Commun. **174** (2006) 87.
[22] Lüscher, M., Comput. Phys. Commun. **165** (2005) 199.
[23] Frommer, A., Nockel, B., Gusken, S., Lippert, T., and Schilling, K., Int. J. Mod. Phys. **C6** (1995) 627.
[24] Jegerlehner, B., Preprint hep-lat/9612014 (1996).
[25] Jegerlehner, B., Nucl. Phys. Proc. Suppl. **63** (1998) 958.